Errata
microSPARC-IIep™ Chip

Tapeout versions of the microSPARC-IIep chip are designated with different JTAG and VA mask IDs, as shown in TABLE 1.

TABLE 1 Tapeout Summary

Tapeout Version	JTAG ID	VA Mask ID	Errata	Remarks
2.0	0x1016d06d	0x34		Initial release
2.1	0x3016d06d	0x36	1	Fixed cache RAMs for improved yield and PCI DMA write bursts of greater than 64 bytes
2.2[1]	0x4016d06d	0x37	2	Fixed PCI satellite mode configuration operations and DMA write problems with 6-byte writes and partial writes with IRDY wait states.

[1]Denotes RTL release with parts not yet released to production

Errata 1

PCI DMA write bursts that are greater than 64 bytes can cause data corruption to memory. A FIFO data pointer within the PCIC is set incorrectly if it exceeds the FIFO full capacity.

Affected RTL Modules

The RTL modules that are affected are as follows:

`slave.v` (`pcic`)	DMA write burst size limits
`dp_mmu.v`	New VA mask ID
`jtag_subblocks.v`	New JTAG ID

Workaround

Program all PCI DMA writes (master devices) to a burst size of 64 bytes or less.

Errata 2

When operating in PCI satellite mode, configuration read and write operations hang the PCI bus, leaving the transfer incomplete.

During 6-byte DMA writes, the data are corrupted, a phenomenon caused by a timing window at startup of the DMA write state machine. (DMA writes of 5, 6, or 7 bytes all can have data corruption.)

During DMA writes with not all byte enables on and IRDY wait states inserted, the DMA data can become corrupted.

Affected RTL Modules

The RTL modules that are affected are as follows:

`config.v` (`pcic`)	Changed port description on `config_access_complete`
`rl_mcb_sm.v`	Changed port description on `sync_t2`
`dp_mmu.v`	New VA mask ID
`jtag_subblocks.v`	New JTAG ID
`afxmaster.v`	Fixed the 6-byte DMA write problem
`afxm_sm.v`	Fixed partial PCI writes with IRDY wait states

Workaround

No existing platforms use the microSPARC-IIep chip in satellite mode. Since the configuration operations hang the PCI bus, do not connect the IDSEL pin if you use it as a slave.

There are no workarounds for the 6-byte or partial DMA writes with IRDY wait states. Avoid these scenarios.

Possible Future Enhancements:

Disable the AFX timeout counter with a PCI control register bit. However, this method has presented problems for some customers when they operate in slave mode and use an external arbiter to implement PCI locking, that is, long latencies to allow PCI PIO reads to get to the PCI bus.

Alternately, implement with a new control bit or use the external arbiter signal.